

LISTING OF CLAIMS

1. (currently amended) A computer-implemented method of creating a compressed memory reference trace for a program said compressed memory reference trace to be stored in memory associated with a computer running said program, said method comprising:

selecting each a sequence of events in a control flow for said program;

obtaining a sequence of values for each of said events, which values were obtained by executing said program;

compressing each said sequence of values to generate a compressed sequence of values for each event, wherein the collection of compressed sequences of values of events generates a compressed memory reference trace;

ordering said values of said compressed memory reference trace according to information in said selected sequence of events in the control flow of said program to generate an uncompressed trace of said program; and

storing at least one of said compressed memory reference trace and said uncompressed ~~uncompressed~~ trace.

2. (currently amended) A method as recited in claim 1, wherein said values of said compressed memory reference trace are ordered in ~~in~~ order of said selected events.

3. (previously presented) A method as recited in claim 1, wherein said sequence of events for said program is selected by dividing said control flow of said program into blocks of instructions and by associating an event with selected instructions in a block.

4. (original) A method as recited in claim 3, wherein said program is divided into said blocks according to the occurrence of a branch instruction, where each block has only one branch instruction which is the last instruction in each said block.

5. (original) A method as recited in claim 1, wherein each said sequence of values for each said event is compressed based upon recognized patterns in each said sequence.

6. (original) A method as recited in claim 5, wherein said recognized patterns comprise at least one of the following patterns: strided patterns and repeat patterns.

7. (original) A method as recited in claim 1, wherein said selected events are branch instructions and wherein values for latter said selected events are branch targets taken by said branch instructions.

8. (currently amended) A method as recited in claim 1 ~~claim 2~~, wherein said information comprises target addresses and wherein said selected events comprise branch instructions.

9. (original) A method as recited in claim 1, further comprising the step of:

using said compressed sequence of values for an event corresponding to a load instruction to pre-fetch values during the execution of a program.

10. (original) A method as recited in claim 1, further comprising the step of:

using said compressed sequence of values for an event corresponding to a branch instruction to perform branch prediction during the execution of a program.

11. (previously presented) A method as recited in claim 1, further comprising:

dividing said compressed memory reference trace into segments, wherein said sequence of compressed values in a segment corresponds to a contiguous sequence of values in said uncompressed trace.

12. (original) A method as recited in claim 11, wherein a segment is terminated at the end of a block such that the size of the segment is between two predetermined values.

13. (currently amended) A program storage device readable by a digital processing apparatus and having a program of instructions which are tangibly embodied on the storage device and which are executable by the processing apparatus to perform a method of creating a compressed memory reference trace for a program for storage of said compressed trace in a memory associated with said processing apparatus, said method comprising:

selecting each a sequence of events for said program;

obtaining a sequence of values for each of said events,
which values were obtained by executing said program;
compressing each said sequence of values to generate a
compressed sequence of values for each event, wherein the
collection of compressed sequences of values of all events
generates a compressed memory reference trace for storage
in a memory associated with said processing apparatus; and
ordering said values of said compressed memory reference
trace to generate an uncompressed trace of said program.

14. (canceled)

15. (currently amended) A ~~An~~ computer apparatus for
creating a compressed memory reference trace of a program,
said apparatus comprising:

means for selecting each ~~a~~ sequence of events for said
program;

means for obtaining a sequence of values for each of said
events, which values were obtained by executing said
program;

means for compressing each said sequence of values to
generate a compressed sequence of values for each event,
wherein the collection of compressed sequences of values of
all events generates a compressed memory reference trace;
means for ordering said values of said compressed memory
reference trace to generate an uncompressed trace of said
program, and

means for storing at least one of said compressed memory
reference trace and said uncompressed trace.

16. (previously presented) The method as recited in claim 1 further comprising the steps of:

analyzing said compressed memory reference trace to identify patterns of program behavior; and utilizing said patterns to optimize program execution.

17. (previously presented) The method as recited in claim 16 wherein said utilizing comprises performing pre-fetching.

18. (previously presented) The method as recited in claim 16 wherein said utilizing comprises performing branch prediction.

19. (previously presented) The program storage device as recited in claim 13 wherein the method further comprises the steps of:

analyzing said compressed memory reference trace to identify patterns of program behavior; and utilizing said patterns to optimize program execution.

20. (previously presented) The apparatus as recited in claim 15 further comprising:
means for analyzing said compressed memory reference trace to identify patterns of program behavior to exploit to optimize program execution.